A New Symmetric Key Cryptographic Algorithm for Unicode Compliant Bangla Characters

Mohammad Zakir Hossain Sarker
East West University
Shaila Rahman
University of Asia Pacific
M Lutfar Rahman
University of Dhaka

Abstract

Cryptography is the discipline that embodies principles, means, and methods for the transformation of data in order to hide its information content, prevent its undetected modification and/or prevent its unauthorized use. Now-a-days most of the organizations are increasingly dependent on electronic data for various purposes. Since data travels electronically the alarming issue is the security. Cryptographic algorithms can play vital role to resolve this issue. There are two basic types of cryptography: Secret/Symmetric Key and Public/Asymmetric Key. Symmetric key algorithms are the quickest and most commonly used type of encryption. Here, a single key is used for both encryption and decryption. This paper describes a new Symmetric Key algorithm which is applied on the Unicode compliant Bangla characters. Algorithms for both encryption and decryption are provided here. The advantages of the new algorithm are also described along with the drawbacks. Since government of Bangladesh is planning to introduce e-Governance this kind of cryptographic algorithm should be applied to secure various important and vital data. As per as literatures are concerned, this kind of work for Unicode compliant Bangla characters has not been done yet i.e. this could be the pioneer work.

Keywords

Cryptography, Encryption, Decryption, Symmetric key Algorithms, Unicode Bangla Characters

1. Introduction

We use security services every day in our workplaces and society has established an intricate set of laws and customs surrounding the use of these security services. For example, if we need to identify someone, we ask him or her to appear in person, perhaps with some credentials. Or, he or she is introduced to us by a common acquaintance. If we need to send a paper document securely, we wrap it in an envelope, a double-envelope, a sealed diplomatic bag, or a strongbox. And, if we need to enforce access to buildings, rooms, facilities, computers or information we do so with locks, keys, combinations and guards. We verify the integrity of paper documents, by checking their signatures and the handwriting. In certain cases, documents are sealed with wax, or stamped, or embossed. Anti-forgery features are used on money/cheques. In the paper world, we authorize transactions like cheques and purchase orders with a signature. Now that we live in a digital world, many of the old Paper World mechanisms are not possible like i) We may never get to meet the recipients of our electronic messages ii) All electronic documents look the same, zeroes and ones are eminently forgeable and iii) We need new services to replace envelopes, locks and combinations. Not only our messages and files need new security mechanisms, but the security mechanisms themselves may require additional security mechanisms. Two main security mechanisms are used to provide the digital equivalents of the Paper World security services: Cryptography and Digital Signature. This paper deals only with Cryptography.

S. William [25] stated that, "Cryptography" derives from the Greek word *kruptos,* meaning "hidden". The key to hiding data is to devise a hiding (encryption) mechanism that is very difficult to reverse (i.e., to find the original data) without using the decryption key. Usually, the harder it is to discover the key, the more secure the mechanism. In symmetric (also called "secret-key" and, unfortunately, "private key") encryption, the same key (or another key fairly easily computed from the first) is used for both encryption and decryption. In asymmetric (also called "public-key") encryption, one key is used for encryption and another for decryption. More specifically, this paper deals with the Symmetric Key cryptography. A new Symmetric Key cryptographic algorithm has been propsed in this paper with its advantages and disadvantages. This algorithm has been applied on Unicode compliant Bangla characters.

This research is basically based on literature study. Initially to understand the basics of cryptography we have studied few books like S.

encryption algorithms use the same key for encryption and decryption, while asymmetric algorithms use a public/private key pair.

- *Data integrity:* To ensure data is protected from accidental or deliberate (malicious) modification. Integrity is usually provided by message authentication codes or hashes. A hash value is a fixed length numeric value derived from a sequence of data. Hash values are used to verify the integrity of data sent through insecure channels. The hash value of received data is compared to the hash value of the data as it was sent to determine if the data was altered.

- *Authentication:* To assure that data originates from a particular party. Digital certificates are used to provide authentication. Digital signatures are usually applied to hash values as these are significantly smaller than the source data that they represent.

3. Types of Cryptographic Algorithms

There are several ways of classifying cryptographic algorithms. In this paper, they will be categorized mainly based on the number of keys that are employed for encryption and decryption. The three types of algorithms that will be discussed (according to S. Hebert [13]) are:

- Secret Key Cryptography: Uses a single key for both encryption and decryption

- Public Key Cryptography: Uses one key for encryption and another for decryption

- Hash Functions: Uses a mathematical transformation to irreversibly "encrypt" information

3.1. Secret Key Cryptography

In secret key cryptography, a single key is used for both encryption and decryption. As shown in Figure 1, the sender uses the key (or some set of rules) to encrypt the plaintext and sends the cipher-text to the receiver. The receiver applies the same key (or rule-set) to decrypt the message and recover the plaintext. Because a single key is used for both functions, secret key cryptography is also called symmetric encryption.

William [25], B. Schneier [23], W. Michael [22], N. Kibitz [19] etc. and then gone through few proceedings and journal papers, articles from the Internet like S. M. Bellovin and M. Merritt [7], C. David [10], R. Anderson and R. Needham [2], K. Gary [11], S. Goldwasser and S. Micali [12], S. Hebert [13], D. Jhon [18] etc. After learning the basics of cryptography we have studied few renowned cryptographic algorithms in detail. These are: Data Encryption Standard (DES), Triple DES, Advanced Encryption Standard (AES), RC2 , RC5, Rivest-Shamir-Adleman (RSA) etc. We have studied these algorithms very rigorously from various sources like P. Barrett [3], W. Hohl, X. Lai, T. Meier and C. Waldvogel [14], M. I. Jabiullah, S. M. Rahman, M. L. Rahman and M. A. Hossain [16], D. Jablon [17], S. Mahmud, S. K. Dey and M. L. Rahman [20], F. Matthew [21], M. I. Sharif, E. Karim, A.N. Mahmood and M. A. Mottalib [24], an Internet article [28], N. Alexandris, M. Burmester, V. Chrissikopoulos and Y. Desmedt [1], M. Bellare, R. Canetti and H. Krawczyk [4], M. Bellare, O. Goldreich and S. Goldwasser [5], M. J. Beller, L. F. Chang and Y. Yacobi [6], S. Blackburn, S. Murphy and J. Stern [8] etc. As our aim of this work was to develop cryptographic algorithms for Bangla Unicode Characters, so after having the deep knowledge in few cryptographic algorithms we have studied various articles, papers and few other resources (i.e. M. Hossain [15], [26],[27], [31] etc.) to know the detail about Unicode compliant Bangla characters. Then we have developed our new algorithm which is described in this paper. As far as literature is concerned this could be the pioneer work in the field of Cryptography for Unicode compliant Bangla characters.

2. The Purpose of Cryptography

In a typical situation where cryptography is used, two parties (X and Y) communicate over an insecure channel. X and Y want to ensure that their communication remains incomprehensible by anyone who might be listening. Furthermore, because X and Y are in remote locations, X must be sure that the information she receives from Y has not been modified by anyone during transmission. In addition, she must be sure that the information really does originate from Y and not someone impersonating Y. According to K. Gary [11] Cryptography is used to achieve the following goals:

- *Confidentiality:* To ensure data remains private. Confidentiality is usually achieved using encryption. Encryption algorithms (that use encryption keys) are used to convert plain text into cipher text and the equivalent decryption algorithm is used to convert the cipher text back to plain text. Symmetric

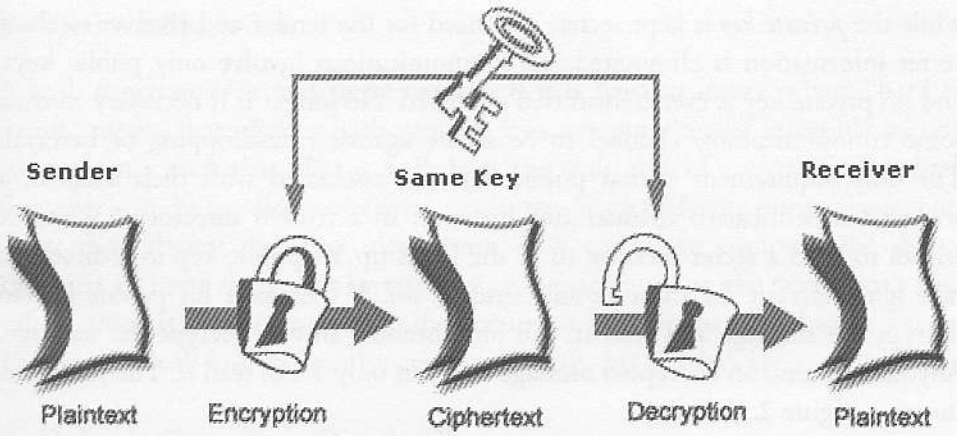

Figure 1: Secret Key Cryptography

With this form of cryptography, it is obvious that the key must be known to both the sender and the receiver. The biggest difficulty with this approach, of course, is the distribution of the key. We have to secure the key from access by unauthorized agents because any party that has the key can use it to decrypt data. Secret key cryptography schemes are generally categorized as being either stream ciphers or block ciphers. Stream ciphers operate on a single bit at a time, and implement some form of feedback mechanism so that the key is constantly changing. A block cipher is so-called because the scheme encrypts one block of data at a time using the same key on each block.

3.2. Public Key Cryptography

The main problem of Secret Key Cryptography is getting the sender and receiver to agree on the secret key without anyone else finding out. If they are in separate physical locations, they must trust a courier, or a phone system, or some other transmission medium to prevent the disclosure of the secret key being communicated. Anyone who overhears or intercepts the key in transit can later read, modify, and forge all messages encrypted or authenticated using that key. The generation, transmission and storage of keys are called key management; all cryptosystems must deal with key management issues. Because all keys in a secret-key cryptosystem must remain secret, secret-key cryptography often has difficulty providing secure key management, especially in open systems with a large number of users. The concept of Public *Key Cryptography* was introduced in 1976 by Whitfield Diffie and Martin Hellman [9] in order to solve the key management problem. In their concept, each person gets a pair of keys, one called the Public Key and the other called the Private Key. Each person's *public key* is published

while the *private key* is kept secret. The need for the sender and receiver to share secret information is eliminated; all communications involve only public keys, and no private key is ever transmitted or shared. No longer is it necessary to trust some communications channel to be secure against eavesdropping or betrayal. The only requirement is that public keys are associated with their users in a trusted (authenticated) manner (for instance, in a trusted directory). When X wishes to send a secret message to Y, she looks up Y's public key in a directory, uses it to encrypt the message and sends it off. Y then uses his private key to decrypt the message and read it. No one listening in can decrypt the message. Anyone can send an encrypted message to Y but only Y can read it. The process is shown in figure 2.

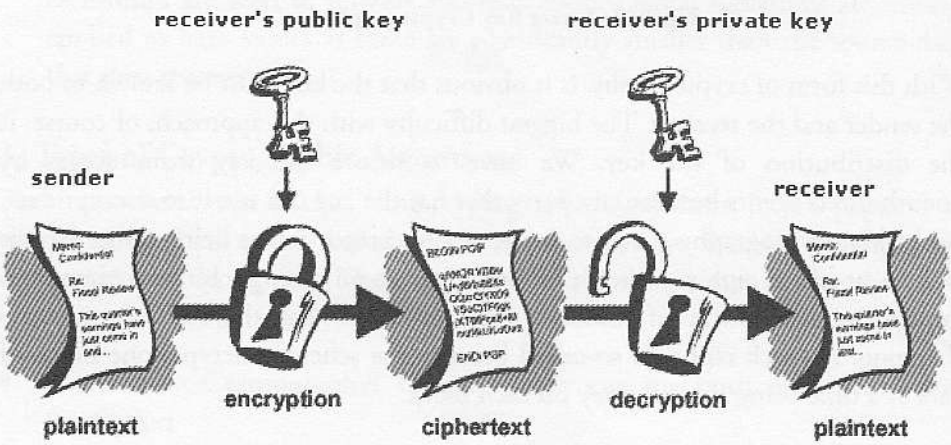

Figure 2: Public Key Cryptography

3.3. Hash Functions

A hash function H is a transformation that takes a variable-size input m and returns a fixed-size string, which is called the hash value h (that is, $h = H(m)$). Hash functions with just this property have a variety of general computational uses, but when employed in cryptography the hash functions are usually chosen to have some additional properties [30].

The basic requirements for a cryptographic hash function are:

- the input can be of any length,
- the output has a fixed length,
- $H(x)$ is relatively easy to compute for any given x,
- $H(x)$ is one-way,

- $H(x)$ is collision-free.

A hash function H is said to be *one-way* if it is hard to invert, where "hard to invert" means that given a hash value h, it is computationally infeasible to find some input x such that $H(x) = h$. Perhaps the main role of a cryptographic hash function is in the provision of digital signatures. Since hash functions are generally faster than digital signature algorithms, it is typical to compute the digital signature to some document by computing the signature on the document's hash value, which is small compared to the document itself, known as Message Digest. Examples of well-known hash functions are MD2, MD5, SHA etc.

4. Unicode Compliant Bangla Characters

Fundamentally, computers just deal with numbers. They store letters and other characters by assigning a number for each one. Initially, there were hundreds of different encoding systems for assigning these numbers. No single encoding could contain enough characters: for example, the European Union alone requires several different encodings to cover all its languages. Even for a single language like English no single encoding was adequate for all the letters, punctuation, and technical symbols in common use. To solve this problem the Unicode Consortium which is a non-profitable organization was founded. Its objective is to develop, extend and promote use of the Unicode Standard [29].

Encryption and decryption of bangla characters requires representing each bangla character into a specific code. Since there is no widely accepted national standard Bangla character code set before 2004, different character code sets are made and used by several software companies like Proshika, Bijoy, Sulekha, Lakhani etc. Bangladesh Standard Code for Information Interchange (BDS 1520: 1995) has been accepted by Bangladesh Standards and Testing Institution (BSTI) on 15th July, 1995 [26]. This code set includes complete set of Bangla characters and symbols including conjunctive characters and in order to send data, the necessary control characters and their codes are kept similar to ISO Latin code set. In the year 2000, this standard was revised by the computer Related National Committee and known as BDS 1520:2000 [27]. According to M. Hossain [15], it contains compound characters and ligatures that are contradictory to the principle of Unicode. To get rid of all of these problems, The "Keyboard Promitokaran Committee, Bangladesh" has been formed in the year of 2003. This committee has proposed a four layer keyboard layout and Bangla character code set which is fully Unicode compliant. In the figure 3, the Bangla Unicode character set and their corresponding Unicode value is shown: [31]

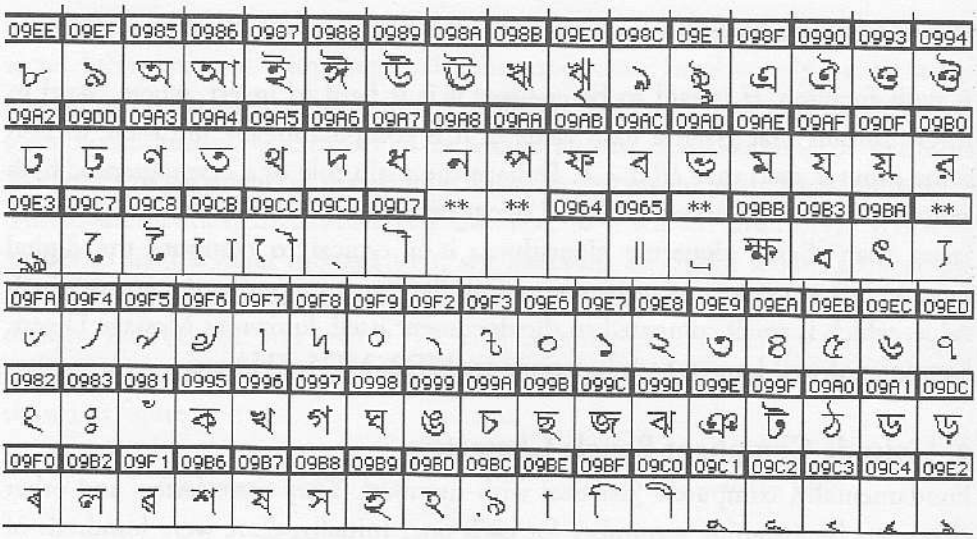

Figure 3: Unicode Compliant Bangla Character Set (Proposed by the Keyboard Promitokaran Committee, Bangladesh)

5. New Symmetric Key Cryptographic Algorithm for Unicode compliant Bangla

In this system, cryptographic algorithms (both Encryption and Decryption) have to apply characterwise. Here, sender has to use two keys. One is the "Key to XOR" and other is the "Private Key". On the other hand, receiver has to know both the keys to decrypt the encrypted text or cipher text. Figure 4 will explain the entire process clearly. So far we havn't think about the key distribution mechanism since our initial goal was to design an algorithm for the Unicode compliant Bangla characters.

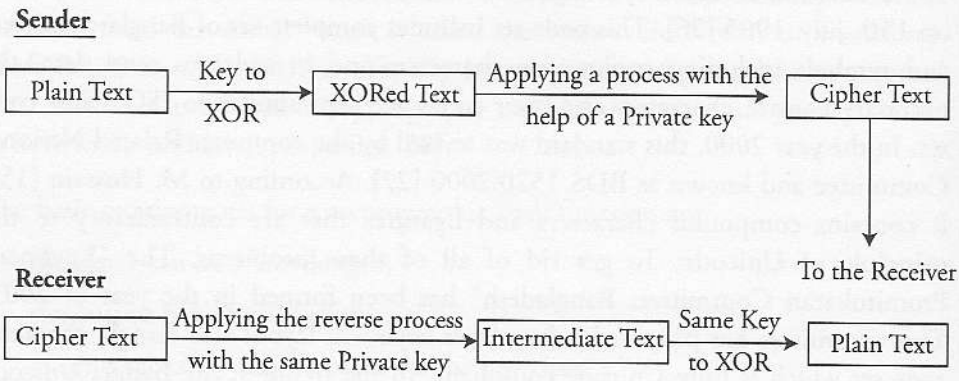

Figure 4: Depiction of the Whole System

5.1. Encryption Algorithm

Step 1: Find out the Unicode value (in binary) of the Bangla character.

Step 2: As we know, each of the Unicode compliant Bangla character is represented by a 16 bit binary number, we must use that 16 bit number in our algorithm. We have to put some leading zeros if it is required. A for example, the Unicode value of is অ 0985 in hex, 2437 in decimal and 100110000101 in binary. According to our algorithm we have to put 4 (four) 0s (zero) in the left hand side to make the value of the character অ as: 0000100110000101

Step 3: Take any 16 bit key as the "Key to XOR". XOR the Bangla character using this key to get the XORed character.

Step 4: Take an 8 bit number as the "Private Key" which must be >=10000000

Step 5: Now, divide the XORed character with the private key. After dividing we will have a 9 bit (maximum) Quotient and 7 bit (maximum) Remainder, no matter whatever the character is. We have to put some leading 0s (zero) if the quotient and remainder is less than 9 and 7 bits respectively.

Step 6: Putting 7 bit remainder and than 9 bit quotient from the left hand side i.e. from the Most Significant Bit (MSB) side, we get a 16 bit number which would be the encrypted character i.e. cipher text.

5.2. Decryption Algorithm

Step 1: Take the 16 bit encrypted character.

Step 2: Multiply the 9 bit quotient by the "Private Key" used while encrypting. Then, add the 7 bit remainder with this.

Step 3: Put required number of leading 0s (zero) to make the number (produced in step 2) a 16 bit number.

Step 4: XOR the number with the same "Key to XOR" used while encrypting.

Step 5: The number which is generated in step 4 would be the Unicode value of the original character.

5.3. A Case Study

Let's take অ as the original Bangla character. Its Unicode value is 0985 in hex, 2437 in decimal and 100110000101 in binary. Since the binary value is 12 bits lets add 4 leading 0s. So it would become 0000100110000101. Now let's take 1000100010001000 as the "key to XOR". After XORing 0000100110000101 with 1000100010001000 we will get 1000000100001101 as the XORed character. Let's use 10001000 as the private key. After dividing 1000000100001101 i.e. XORed character by 10001000 i.e. private key we will get 11110010 as the quotient and 1111101 as the remainder. As the quotient is not a 9 bit number we have to make it a 9 bit number by putting a 0 in the left hand side. So it becomes 011110010. Now putting 7 bit remainder and then 9 bit quotient from the left hand side we will get the encrypted character. For this case which would be 1111101011110010. In decimal it is 64242 and in hex FAF2. Corresponding character of this value is □. Table 1 shows process described above.

Table 1: Processes in the Sender's End using the Encryption Algorithm

Plain Text	Key to XOR	XORed Text	After dividing XORed Text by the private key 10001000		Cipher Text
			Remainder	Quotient	
অ (0000100110000101)	1000100010001000	1000000100001101	1111101	011110010	□(1111101011110010)

In the receiver-end, 9 bit quotient of the cipher text 1111101011110010 i.e. 011110010 would be multiplied by the private key used by the sender which is 10001000. So after multiplying 011110010 by 10001000 we get 1000000010010000. Now, after adding the 7 bit remainder with this we have 1000000100001101. Finally, the receiver will get the original text by XORing this value with the "Key to XOR" used by the sender i.e. 1000100010001000. So, 1000000100001101 XOR 1000100010001000 is equivalent to 0000100110000101 i.e. অ. Table 2 will explain the entire process.

Table 2: Processes in the Receiver's End using the Decryption Algorithm

Cipher Text	After Multiplying 9 bit Quotient by the Private key 10001000	After adding 7 bit remainder	Key to XOR	After XORing
□(11110101111010)	1000000010010000	1000000100001101	1000100010001000	অ(0000100110000101)

In this way we can encrypt all the alphabets available in a Bangla text file one by one.

6. Advantages of the System

Advantages of this system are described below.

a. It would not be cost-effective to use the various existing algorithms for a small amount of data. This system will work very smoothly and would be cost-effective in such case.

b. It is always more difficult to guess 2 (two) keys than 1 (one) key. That's why this system more secured since 2 (two) keys are used here.

c. This system would be faster since the internal design is simpler.

7. Drawbacks of the System

There are few drawbacks of the system also. These are pointed out below.

a. Since the algorithms are working characterwise the performance would not be lucrative for large amount of data.

b. Key-length is smaller compare to few other existing algorithms.

8. Conclusion

Cryptography is used to achieve few goals like Confidentiality, Data integrity, Authentication etc. of the data which has sent to the receiver from the sender. Now, in order to achieve these goals various cryptographic algorithms are developed by various people. It has been found that the algorithms which are available at this moment are mostly used for English language and those algorithms are more or less difficult or complex in nature, and of-course it is quite obvious since those algorithms are used to maintain high level of security against any kind of forgeries. For a very minimal amount of data those algorithms wouldn't be cost effective since those are not designed for small amount of data. The aim of this work was to design and implement a new algorithm for Bangla language and also to design a cost-effective algorithm to encrypt a small amount of data The algorithms that are discussed here have been designed in a quite simple manner (but of-course not sacrificing the security issues) to make it cost-

effective for small amount of data. As per as literatures are concerned, no extensive work has been done so far to apply cryptographic algorithms in Unicode compliant Bangla characters. That's why we could not compare this algorithm with any existing one. We havn't yet found out the breaking probability of our proposed algorithm. We will try to find out the breaking probability in our next phase of work. Here, a single key is used for both encryption and decryption i.e. it is fallen under secret key cryptographic algorithm. But as public key cryptography is more secured then secret key cryptography our next task would be to develop and design a public key cryptographic algorithm in a simple manner as it is done in this paper.

References

[1] N. Alexandris, M. Burmester, V. Chrissikopoulos and Y. Desmedt, A secure key distribution system , *Proceedings of the 3rd Symposium on State and Progress of Research in Cryptography,* Rome, Italy, 1993, pp 30–34

[2] R. Anderson and R. Needham, Robustness principles for public key protocols, *Advance in Cryptology–CRYPTO '95 (LNCS 963),* 1995, pp 236–247

[3] P. Barrett, Implementing the Rivest Shamir and Adleman public key encryption algorithm on a standard digital signal processor, *Advances in Cryptology–CRYPTO '86 (LNCS 263),* 1987, pp 311–323

[4] M. Bellare, R. Canetti and H. Krawczyk, Keying hash functions for message authenticaion, *Advances in Cryptology–CRYPTO '96 (LNCS 1109),* 1996, pp 1–15

[5] M. Bellare, O. Goldreich and S. Goldwasser, Incremental cryptography: The case of hashing and signing, *Advances in Cryptology–CRYPTO '94 (LNCS 839),* 1994, pp 216–233

[6] M.J. Beller, L. F. Chang and Y. Yacobi, Security for personal communications services: public-key vs. private key approaches, *The Third IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'92),* 1992, pp 26–31

[7] S. M. Bellovin and M. Merritt, Augmented Encrypted Key Exchange a Password Based Protocol Secure Against Dictionary Attacks and Password File Compromise, *Proceedings of the First ACM Conference on Computer and*

Communications Security, 1993, pp 243-250

[8] S. Blackburn, S. Murphy and J. Stern, The cryptanalysis of a public-key implementation of finite group mappings, *Journal of Cryptology,* 1995, pp 157–166

[9] C. David, A Review of the Diffie-Hellman Algorithm and its Use in Secure Internet Protocols, an article available at http://www.sans.org/rr/papers/20/751.pdf last accessed on January 16, 2006

[10] W. Ford and M. Baum, *Secure Electronic Commerce: Building the Infrastructure for Digital Signatures and Encryption,* 2nd Edition, 2001, Prentice Hall

[11] K. Gary, An Overview of Cryptography, an article available at www.garykessler.net/library/crypto.html last accessed on January 17, 2006

[12] S. Goldwasser and S. Micali, Probabilistic Encryption, *Journal of Computer and System Sciences,* 1999, Vol 28, pp 270-299

[13] S. Hebert, A Brief History of Cryptography, an article available at http://cybercrimes.net/Cryptography/Articles/Hebert.html last accessed on January 15, 2006

[14] W. Hohl, X. Lai, T. Meier and C. Waldvogel, Security of iterated hash functions based on block ciphers, *Advances in Cryptology–CRYPTO '93 (LNCS 773),* 1993, pp 379–390

[15] M. Hossain, Unicode, *Computer Tomorrow,* Vol 4 No 8, June, 2003 pp 21-35

[16] M. I. Jabiullah, S. M. Rahman, M. L. Rahman and M. A. Hossain, Secure Pseudoorandom Bit Generation for Cryptographic Application , *Proceedings of International Conference on Computer and Information Technology (ICCIT),* Dhaka, Bangladesh, 2001 pp 275-277

[17] D. Jablon, Strong Password Only Authenticated Key Exchange, *Computer Communication Review, ACM SIGCOMM,* Vol 26 No 5, 1996, pp 5-26

[18] D. Jhon, Digital Signatures and Encryption, an article available at www.anr.state.vt.us/isp/digsig.htm last accessed on November 14, 2005

[19] N. Kibitz, A *Course in Number Theory and Cryptography,* 2nd Edition, 1994,

Springer-Verlag

[20] S. Mahmud, S. K. Dey and M. L. Rahman, Implementation of Public Key Encryption Using RSA Algorithm for Bangla , *Proceedings of National Conference on Computer Processing of Bangla (NCCPB) 2004,* Independent University, Dhaka, Bangladesh, pp 225 – 231

[21] F. Matthew, How to implement the Data Encryption Standard (DES), a step by step tutorial Version 1.24 available at www.hack.gr/users/sd/documents/cryptography/des-algorithm-details.txt last accessed on November 7, 2005

[22] W. Michael, *Cryptography in C/C++*, 2nd Edition, 2001, Apress Publisher

[23] B. Schneier, *Applied Crytography,* 2nd Edition, Jhon Wily & Sons, 1996

[24] M. I. Sharif, E. Karim, A. N. Mahmood and M.A. Mottalib, Another Tip for Secure RSA Key Selection, *Proceedings of International Conference on Computer and Information Technology (ICCIT),* Dhaka, 2001, pp 283-285

[25] S. William, *Cryptography and Network Security: Principles and Practice,* 2nd edition, 1999, Prentice-Hall, Inc.

[26] Bangladesh Standard Bangla Coded Character Set , Bangladesh Standards and Testing Institution (BSTI), BDS 1520: 1995, 1995

[27] Bangladesh Standard Specification for Bangla Coded Character Set for Information Interchange (First Revision) , Bangladesh Standards and Testing Institution (BSTI), BDS 1520: 2000, 2000

[28] Basic Cryptographic Algorithms, an article available at www.itsc.state.md.us/oldsite/info/InternetSecurity/Crypto/CryptoIntro.htm#Algorithms last accessed on November 6, 2005

[29] http://www.unicode.org/faq last accessed on January 20, 2006

[30] Hash Function and Block Cipher Information at burtleburtle.net/bob/hash last accessed on January 14, 2006

[31] Report of the Standardization Committee of Bangla Keyboard in Computer (Bangla Keyboard Promitokaran Committee), Bangladesh Computer Council, Dhaka, Bangladesh